

Cutting multi-control quantum gates with ZX calculus

Christian Ufrecht, Maniraman Periyasamy, Sebastian Rietsch, Daniel D. Scherer, Axel Plinge, and Christopher Mutschler

Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, Division Positioning and Networks, Nuremberg, Germany

Circuit cutting, the decomposition of a quantum circuit into independent partitions, has become a promising avenue towards experiments with larger quantum circuits in the noisy-intermediate scale quantum (NISQ) era. While previous work focused on cutting qubit wires or two-qubit gates, in this work we introduce a method for cutting multi-controlled Z gates. We construct a decomposition and prove the upper bound $\mathcal{O}(6^{2K})$ on the associated sampling overhead, where K is the number of cuts in the circuit. This bound is independent of the number of control qubits but can be further reduced to $\mathcal{O}(4.5^{2K})$ for the special case of CCZ gates. Furthermore, we evaluate our proposal on IBM hardware and experimentally show noise resilience due to the strong reduction of CNOT gates in the cut circuits.

1 Introduction

Quantum computing [1] in the noisy-intermediate scale quantum (NISQ) era is limited by the strong impact of noise and the small number of available qubits [2]. As a result, current hardware is far from being able to execute quantum algorithms with provable quantum advantage, such as Shor’s [3] or Grover’s [4] algorithm. To overcome these hardware restrictions to some extent, circuit-cutting techniques have recently attracted a lot of attention. When scaling up problem instances for the search of empirical quantum advantage in quantum-machine learning tasks or for the quantum approximate optimization algorithm (QAOA) [5], these methods are expected to become relevant tools and will likely be an inherent part of near-term quantum software frameworks [6].

Consider a quantum circuit consisting of two partitions, only connected by a few wires or two-qubit gates. By decomposing the identity channel, Peng et al. [7] introduced *wire cutting* where a qubit line is cut along the direction of time. This method was subsequently further investigated [8, 9, 10, 11] with respect to the interplay between circuit cutting and noise [12, 13, 14, 15], automatic allocation of classical and quantum computational resources [16, 17] and compilation [18]. Similarly, Mitarai and Fujii [19, 20, 21]

Christian Ufrecht: christian.ufrecht@iis.fraunhofer.de

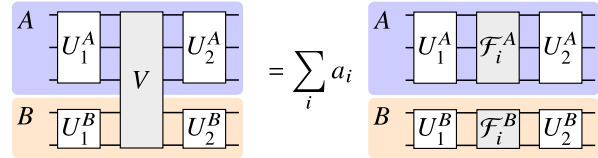


Figure 1: Consider the quantum circuit shown on the left. It is a circuit with five qubits represented by the horizontal lines, also referred to as wires. Time evolution of the initial state is illustrated by the white and grey boxes where time flows from the left to the right. They represent quantum gates, that is, unitary transformations on subsets of the qubits. The circuit contains a single gate V that connects the two partitions A and B that are otherwise independent. After decomposition of the quantum channel corresponding to V , the circuit disintegrates into a weighted sum over independent circuit pairs, which contain quantum gates or measurements denoted by \mathcal{F}_i^A and \mathcal{F}_i^B . Here, we employ the calligraphic notation otherwise reserved for superoperators to indicate that the gate or projectors are determined from the superoperator decomposition. The result of the original circuit can be restored by evaluating sequentially each of the circuits on possibly smaller quantum devices. Note that the equality has to be understood on the superoperator rather than on the gate level.

proposed *gate cutting*, the direct decomposition of a unitary channel corresponding to a two-qubit gate. As in probabilistic error mitigation [22, 23, 24, 21] the variance of the estimator for the quantity to be measured increases [25] in case of a cut circuit. As a consequence, all variants of circuit cutting come with a constant κ , characterizing the sampling overhead, the factor $\mathcal{O}(\kappa^2)$ of more samples required to estimate the decomposed circuit to the same accuracy as the original one. If K cuts are performed, the sampling overhead increases to $\mathcal{O}(\kappa^{2K})$. Of course, the exponential overhead is in line with our expectation of classical hardness for simulation of general quantum circuits. Piveteau et al. [26] showed that this overhead can be significantly reduced when a common decomposition of several Bell states with subsequent gate teleportation is performed. Ref. [26] also provides optimal decompositions for several important two-qubit gate types based on the robustness of entanglement measure [27]. A strong reduction of sampling overhead can also be achieved for joint cutting of wires as shown by Lowe et al. [28].

In this work, we provide explicit decompositions of

Table 1: Overview of recent findings on CZ and wire cutting. We contrast the sampling overhead for K cuts associated with different methods and compare them to the main results of the present work.

	sampling overhead	joint cutting
CZ gate	$\mathcal{O}(3^{2K})$ [20]	$\mathcal{O}(4^K)$ [26]
wire	$\mathcal{O}(4^{2K})$ [7]	$\mathcal{O}(4^K)$ [28]
CCZ gate	$\mathcal{O}(4.5^{2K})$ [this work]	–
MCZ gate	$\mathcal{O}(6^{2K})$ [this work]	–

multi-controlled Z (MCZ) gates with the help of ZX-calculus, a tensor-network description for quantum circuits. After explaining the ideas underlying gate-cutting in more detail and comparing the sampling overhead of different methods in Sec. 2, we prove the main result of this article, the general decomposition of an MCZ gate in Sec. 3 with the help of ZX calculus. In Sec. 4, we evaluate the sampling complexity associated with the decomposition of an MCZ gate. We show the upper bound $\kappa = 6$, which, remarkably, is independent of the number of control qubits. For a CCZ gate, we find the smaller value $\kappa = 4.5$. We conclude this article with experimental results on the IBM Q system Ehningen discussed in Sec. 5. We observe a strong reduction of the CNOT-gate count in the cut circuits and, therefore, resilience to noise. As a particular application of MCZ-gate cutting, we anticipate its use in the alternating operator ansatz [29, 30], a variant of the QAOA [5] algorithm for constrained optimization. Another promising application is the simulation of MCZ gates connecting qubits far apart on the hardware graph as done for two-qubit gates in Refs. [20, 31].

2 Cutting a quantum gate

Consider a quantum circuit where the qubits are grouped into two partitions A and B , only connected by one possibly multi-qubit gate V as shown in Fig. 1. Furthermore, assume the factorizing initial state $\rho = \rho^A \otimes \rho^B$ and observable $O = O^A \otimes O^B$ where the superscript labels the partition. These assumptions are, for example, satisfied when all qubits are initialized to $|0\rangle$ and a Pauli string is measured. Circuit cutting, also referred to as circuit decomposition, circuit fragmentation or circuit knitting, is the task of finding a decomposition of the unitary channel \mathcal{V} corresponding to the gate V so that

$$\mathcal{V} = \sum_i a_i \mathcal{F}_i \quad (1)$$

where $\mathcal{F}_i(\rho^A \otimes \rho^B) = \mathcal{F}_i^A(\rho^A) \otimes \mathcal{F}_i^B(\rho^B)$ are local channels on the partitions A and B as shown in Fig. 1 and

a_i are real expansion coefficients. For \mathcal{U}_1 corresponding to $U_1^A \otimes U_1^B$ and \mathcal{U}_2 corresponding to $U_2^A \otimes U_2^B$, this decomposition allows us to determine the expectation value of the observable as

$$\langle O \rangle = \text{tr}(O \mathcal{U}_2 \circ \mathcal{V} \circ \mathcal{U}_1(\rho)) \quad (2)$$

$$= \sum_i a_i \text{tr}(O^A \otimes O^B \mathcal{U}_2 \circ \mathcal{F}_i \circ \mathcal{U}_1(\rho^A \otimes \rho^B)) \quad (3)$$

$$= \sum_i a_i \langle O^A \rangle_i \langle O^B \rangle_i. \quad (4)$$

In Eq. (4) we denote by $\langle \cdot \rangle_i$ the expectation value with respect to the state on partition A and B , evolved by \mathcal{F}_i^A and \mathcal{F}_i^B , respectively, that is

$$\langle O^\alpha \rangle_i = \text{tr}(O^\alpha \mathcal{U}_2^\alpha \circ \mathcal{F}_i^\alpha \circ \mathcal{U}_1^\alpha(\rho^\alpha)) \quad (5)$$

for $\alpha = A, B$. If \mathcal{F}_i^α are unitary channels or measurement operations themselves, each term in the sum in Eq. (4) can be evaluated on a quantum computer. In the case of a more general quantum circuit, when all gates connecting the two partitions are cut, they become independent and the quantum circuits can be evaluated sequentially on a smaller device. We emphasize that the term circuit cutting as used here deals with decompositions of quantum circuits at the level of superoperators, rather than at the level of unitaries as in Ref. [32]. The output of a single experimental shot on a quantum computer is typically a bitstring. The expectation value of an observable is then obtained by a classical post-processing function on the bitstrings of multiple runs. Modeling the outcome of each experimental run as i.i.d. random variables, the number of required samples to achieve a given additive error is determined via the variance of the post-processing function. When a circuit is cut, the variance of the modified estimator for $\langle O \rangle$ via Eq. (4) increases. Consequently, more experimental runs are required to estimate the result of the original circuit to the same given additive error. More specifically, this sampling overhead is exponential (in the number of cuts). The parameter

$$\kappa = \sum_i |a_i| \quad (6)$$

then quantifies the sampling overhead $\mathcal{O}(\kappa^2)$ [25]. For completeness, this scaling behavior is re-derived in Appendix E. Eq. (1) is therefore optimal if the \mathcal{F}_i are chosen such that the 1-norm of the vector containing the coefficients a_i is minimal. For K cuts the sampling overhead is $\mathcal{O}(\kappa^{2K})$, however, joint cutting of multiple gates or wires leads to much smaller bounds for some gate types [26, 28]. Tab. 1 summarizes recent findings for cutting of CZ gates and wire cutting and compares the sampling complexities to the main results of the present work. In this work we cut multi-qubit controlled Z gates and provide upper bounds for κ . As we will explore in the next section, ZX calculus is particularly suited for this task.

3 ZX-calculus for circuit cutting

ZX calculus [33, 34] is a tensor-network representation for quantum circuits that together with powerful transformation rules allows diagrammatic reasoning. Since ZX-calculus has been reviewed elsewhere [35], we will only introduce those diagram types and transformation rules necessary for this article. The basic diagrams are Z-spiders, defined as

$$m \text{ --- } \textcircled{\alpha} \text{ --- } n = \underbrace{|0\dots 0\rangle}_n \underbrace{\langle 0\dots 0|}_m + e^{i\alpha} |1\dots 1\rangle \langle 1\dots 1| \quad (7)$$

and X-spiders

$$m \text{ --- } \textcircled{\alpha} \text{ --- } n = \underbrace{|+\dots+\rangle}_n \underbrace{\langle +\dots+|}_m + e^{i\alpha} |-\dots-\rangle \langle -\dots-| \quad (8)$$

where $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$. For $\alpha = 0$, the inset is commonly disregarded. A third tensor, a so-called H-box, is defined as

$$m \text{ --- } \square \text{ --- } n = \sum (-1)^{i_1 \dots i_m j_1 \dots j_n} |j_1 \dots j_n\rangle \langle i_1 \dots i_m| \quad (9)$$

where the sum runs over all $i_1, \dots, i_m, j_1, \dots, j_n \in \{0, 1\}$. Spiders and H-boxes are therefore maps from m - to (un-normalized) n -qubit states, signified by the number of wires ending at the right and the left of the diagrams. When representing ZX diagrams as matrices, we will implicitly assume the computational basis. Then, for example, an H-box is a matrix filled with ones but a minus one in the lower right corner. H-boxes can be viewed as generalized Hadamard gates since

$$-\square = \sqrt{2} \text{---} \boxed{H} \text{---} \quad (10)$$

As apparent from the definitions, spiders and H-boxes correspond to symmetric tensors in all indices. Any quantum circuit can be represented as a ZX diagram, the reverse statement, however, is incorrect since a ZX diagram not necessarily corresponds to a unitary matrix. To obtain the Hermitian conjugate of a ZX diagram, we move all wires ending at the left to the right and those ending at the right to the left and replace all angles with their negative values. In the following we show how to use ZX calculus to decompose the unitary channel corresponding to an MCZ gate into a sum over unitary and measurement channels. Decomposing circuits into simpler parts using ZX calculus has been done before in Refs. [36, 37].

An MCZ gate enjoys a simple representation [38]

$$\text{---} \text{---} \text{---} \text{---} \text{---} = \square \text{---} \text{---} \text{---} \text{---} \text{---} \quad (11)$$

in terms of an H-box with zero wires on the left as shown in Appendix A. In matrix representation,

$MCZ = \text{diag}(1, \dots, 1, -1)$ and we will refer to the number of qubits involved in the MCZ gate as the *order* of the gate. Also note that there is no difference between the control and target qubits in an MCZ gate. The decomposition constructed in this article is based on the H-box fusion rule [38]

$$\square \text{---} \text{---} \text{---} \text{---} \text{---} = \frac{1}{2} \left(\begin{array}{c} \square \text{---} \text{---} \text{---} \text{---} \text{---} \\ \square \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \right), \quad (12)$$

proved in Appendix B for completeness.

Consequently, with Eq. (12) applied to Eq. (11), the unitary channel action $\mathcal{E}_{MCZ}(\rho) = MCZ\rho MCZ^\dagger$ corresponding to the MCZ gate of order $n+m$ acting on an arbitrary density matrix ρ , takes the form

$$\mathcal{E}_{MCZ}(\rho) = \frac{1}{4} \begin{array}{c} \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \rho \begin{array}{c} \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \quad (13)$$

The two partitions (upper m qubits, lower n qubits) are only connected by the tensor Q shaded in grey, the un-normalized Choi operator of a Hadamard gate. Next, we remove the two remaining connections between the partitions by a rank-one decomposition of Q in terms of vectors factorizing over the two partitions

$$Q = \square \text{---} \square = \begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{pmatrix} \quad (14)$$

$$= \sum_i c_i \begin{array}{c} \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \quad (15)$$

In Eq. (15) we introduced the diagrammatic notation for a two-dimensional vector

$$\mathbf{v} = \sum_{j \in \{0,1\}} v_j |j\rangle = \langle \mathbf{v} | \text{---} = \left(\text{---} | \mathbf{v} \rangle \right)^\dagger \quad (16)$$

where v_j is the j -th component of \mathbf{v} . Therefore, the H-box fusion rule in Eq. (13) reduces the cutting of an MCZ gate to the decomposition of a four-by-four matrix. However, only vectors $\mathbf{w}^{(i)}$ and $\mathbf{u}^{(i)}$ are useful whose contraction with the H-box results in channels that can be evaluated on a quantum computer. In Appendix C we prove the identities

$$\text{---} \text{---} \text{---} \text{---} \text{---} = \sqrt{2} \begin{array}{c} \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \boxed{R_z(\theta)} \quad (17)$$

for $\theta \in [0, 2\pi)$ and

$$\text{---} \text{---} \text{---} \text{---} \text{---} = 2 \begin{array}{c} \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \boxed{P_{1\dots 1}} \quad (18)$$

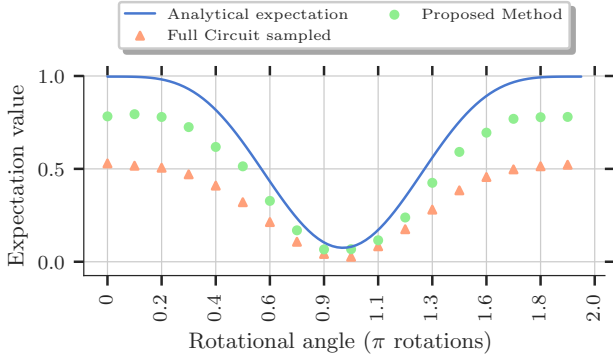


Figure 3: Expectation value estimation of a parameterized circuit containing a 5-qubit MCZ gate on a noisy simulator. Comparison to Fig. 5 shows that the noisy simulation strongly underestimates the influence of noise for sampling the uncut circuit (orange triangles). For the simulations of noise, qiskit’s noise model from `ibmq_ehningen` is used. The noise model is updated based on current calibration and error data in the cloud.

run simulations for 3,4 and 5 qubits and find almost no difference in the standard deviations of the distributions. This behavior is expected since κ is independent from the order of the MCZ gate. Exemplarily, Fig. 2 shows the distribution for random 5-qubit circuits.

In the figure, the white boxes are the interquartile ranges and the bars denote the 5% and 95% quantiles. Fig. 2(1) shows the distribution of the expectation values for the original uncut circuits. As expected, the distribution becomes broader when we instead evaluate the cut circuit in Fig. 2(2). Its standard deviation is 0.0027, which is smaller than the chosen value of $\epsilon = 0.01$. The same effect is observed for $\epsilon = 0.001$ shown in the plots in Fig. 2(3) and Fig. 2(4). Consequently, a tighter version of Eq. (68) to bound the variance might exist. To further validate the proposed method on real quantum hardware and analyze the impact of noise, we performed MCZ gate-cutting experiments on the `ibmq_ehningen` [40] device. The `ibmq_ehningen` device is a super-conducting qubit type quantum hardware with a 27 qubit `ibmq_falcon` processor. The native gate set on this hardware consists of Rotational-Z, Pauli-X, \sqrt{X} , Identity, and CNOT gates. For the experiments, we again generated a random circuit with one free parameter that was scanned in the experiments. The maximum number of shots per job on the `ibmq_ehningen` device is 10^5 and the same was used for each data point, see Supplemental Material for the explicit circuits and a more detailed discussion of the experimental setup. Fig. 4 shows the result for a three-qubit circuit containing a CCZ gate. In the figure, the results obtained by cutting the CCZ gate (green circles) lie slightly closer to the exact curve (blue line) than the results from the uncut circuit (orange triangles).

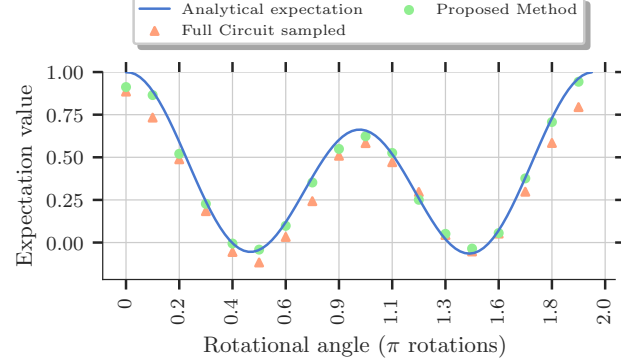


Figure 4: Expectation value estimation of parameterized circuit containing a CCZ gate on `ibmq_ehningen` device. The green circles, sampled from the cut circuits, seem to lie closer to the exact result (blue line) than the result sampled from the uncut circuit (orange triangles). However, more simulations would be needed to verify that cutting a CCZ gate leads to noise reduction for the circuit considered. The x-axis shows the values of the free parameter in the circuit, see Supplementary Material.

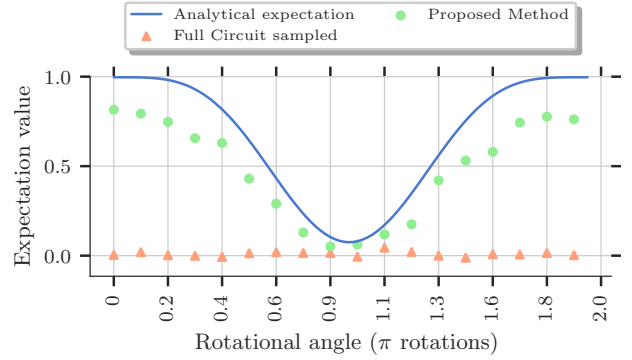


Figure 5: Expectation value estimation of parameterized circuit containing a 5-qubit MCZ gate on the `ibmq_ehningen` device. The green circles, sampled from the cut circuits, still qualitatively match the exact result (blue line). The signal sampled from the uncut circuit (orange triangles) is completely derogated due to the strong influence of noise.

Next, we repeated the experiment with a five-qubit circuit containing an MCZ gate of order five. The result is shown in Fig. 5. From the figure, it is clear that the influence of noise is extreme when executing the transpiled version of the full circuit with an MCZ gate of order five. The resulting expectation values were nothing but a random output centered around zero. However, the expectation value estimated using the proposed method falls much closer to the expectation values obtained by ideal simulation. This resilience to noise can be attributed to the strong reduction of the CNOT-gate count in the cut circuits. While, asymptotically, an MCZ gate of order n can be synthesized with $\mathcal{O}(n)$ CNOT gates with one auxiliary qubit [41], for small n the number of required CNOT gates increases quickly. Consequently, cutting for example

two qubits from an MCZ gate of order five reduces the maximum number of CNOT gates in a circuit to those required to synthesize a double-controlled S gate. This reduction becomes even more pronounced after compiling to topologically restricted hardware. As a consequence, we observe a strong reduction of noise impact. The maximum number of CNOT gates present in the different circuits are shown in Tab. 3. The number of CNOT gates might vary based on the transpilation method used. However, we chose to proceed with the best transpiler offered by the hardware manufacturer to validate our results. As most of the

Table 3: Number of CNOT gates present in the transpiled version of the circuits shown in the Supplementary Material (first column). After circuit cutting, the maximum number of CNOT gates contained in one of the cut circuits reduces considerably (second column).

gate type	CNOT	CNOT cut
CCZ gate	13	3
5-qubit MCZ	114	32

work in the literature is conducted via a simulator, we extended our 5-qubit MCZ experiment shown above to a noisy simulator. The results are shown in Fig. 3. The proposed method still outperforms the full circuit execution even though the noise model provided by the manufacturer does not accurately capture the noise level exhibited by the real device.

6 Conclusion

In this work, we proposed an approach for cutting multi-controlled Z gates by means of ZX-calculus based on the H-box fusion rule. We derived the upper bound $\kappa = 6$ on the sampling overhead that is independent of the order of the gate. We validated the results on IBM hardware and found strong noise reduction due to the reduced amount of CNOT gates in the cut circuit. We anticipate the generalization of our method to multi-controlled rotation gates and extension to multi-qubit rotations. The optimality of the decompositions constructed in this work at present remains an open question.

Acknowledgements

We thank L. Burgholzer and R. Wille for fruitful exchange and discussion.

We acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team.

The research was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy with funds from the Hightech Agenda Bayern

and by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport and Technology through the Center for Analytics-Data-Applications (ADA Center) within the framework of “BAYERN DIGITAL II”.

The research was furthermore funded by the project QuaST, supported by the Federal Ministry for Economic Affairs and Climate Action on the basis of a decision by the German Bundestag.

A Representation of an MCZ gate

In this appendix, we prove the representation of an MCZ gate as stated in Eq. (11). We begin by introducing the diagrammatic notation of a general n -qubit column vector \mathbf{v} as

$$\mathbf{v} = \sum_{i=0}^{2^n-1} v_i |i\rangle = \langle \mathbf{v} | \cdot = \left(\begin{array}{c} \overline{\mathbf{v}} \\ \vdots \\ \mathbf{v} \end{array} \right)^\dagger. \quad (30)$$

With this definition and the Z-spider

$$\text{Z-spider} = \sum_{j \in \{0,1\}} |j\rangle \langle jj|, \quad (31)$$

see Eq. (7), we obtain

$$\begin{array}{c} \text{MCZ gate} \\ \mathbf{v} \end{array} = \left(\sum_{j \in \{0,1\}} |j\rangle \langle jj| \right)^{\otimes n} \sum_{i \in \{0,1\}^n} v_i \bigotimes_{k=1}^n (\mathbb{I} \otimes |i_k\rangle) \quad (32)$$

$$= \sum_i v_i |i\rangle \langle i| = \text{diag}(\mathbf{v}). \quad (33)$$

To guarantee that the correct indices are contracted, we inserted one-qubit identities \mathbb{I} . Consequently, Eq. (33) is proportional to a unitary matrix if all elements of \mathbf{v} have the same absolute value or to a projector on a computational basis state if all but one element of \mathbf{v} are zero. We are now in the position to prove Eq. (11) by setting

$$\langle \mathbf{v} | \cdot = \square \cdot = (1, \dots, 1, -1)^T \quad (34)$$

where we recalled the definition of an H-box from Eq. (9). Consequently, from Eq. (33), we find $\text{diag}(1, \dots, 1, -1)$ for Eq. (11), the matrix representation of an MCZ gate.

B Proof of H-box fusion rule

We proof Eq. (12) from the right to the left. We first define the joint indices $i = (i_1, \dots, i_m)$ and $j = (j_1, \dots, j_n)$ as well as $\pi_i = \prod_{k=1}^m i_k$ and $\pi_j = \prod_{k=1}^n j_k$. We then find in Dirac notation with the indices a, b, a'

and b' running over $\{0, 1\}$, $j \in \{0, 1\}^n$ and $i \in \{0, 1\}^m$

$$\sum_{i,j,a,b} (-1)^{\pi_i a} |i\rangle\langle a| \otimes (-1)^{\pi_j b} |j\rangle\langle b| \sum_{a',b'} (-1)^{a'b'} |a'b'\rangle \quad (35)$$

$$= \sum_{i,j} \sum_{a,b} (-1)^{ab + \pi_i a + \pi_j b} |ij\rangle \quad (36)$$

$$= 2 \sum_{i,j} (-1)^{\pi_i \pi_j} |ij\rangle \quad (37)$$

which is equivalent to the left-hand side of Eq. (12). In the last step we recalled that $\pi_i, \pi_j \in \{0, 1\}$ so that

$$\sum_{a,b} (-1)^{ab + \pi_i a + \pi_j b} \quad (38)$$

$$= (-1)^{-\pi_i \pi_j} \sum_{a,b} (-1)^{(a + \pi_j)(b + \pi_i)} \quad (39)$$

$$= 2(-1)^{\pi_i \pi_j}. \quad (40)$$

Alternatively, the H-box-fusion rule can be derived by the Schmidt decomposition of the vector $(1, \dots, -1)^T$ representing the diagonal elements of the unitary corresponding to the MCZ gate.

C H-box identities

In this appendix, we prove Eq. (17) and Eq. (18). The contraction of a single-qubit vector with an H-box is readily calculated as

$$\langle \mathbf{w} | \text{H-box} \rangle = (w_0 + w_1, \dots, w_0 + w_1, w_0 - w_1)^T \quad (41)$$

by matrix-vector multiplication. Consequently, making use of Eq. (33), we find

$$\langle \mathbf{w} | \text{H-box} \rangle = \text{diag}(w_0 + w_1, \dots, w_0 + w_1, w_0 - w_1). \quad (42)$$

Therefore, Eq. (42) is proportional to a unitary for $|w_0 + w_1| = |w_0 - w_1|$ and to a projector on the state $|1\dots 1\rangle$ for $w_0 = -w_1$. In terms of spiders, these two conditions are either satisfied for $\mathbf{w} = \textcircled{\theta} = \sqrt{2}e^{i\theta/2}(\cos[\theta/2], -i\sin[\theta/2])^T$ or for $\mathbf{w} = \textcircled{\pi} = (1, -1)^T$. The former results in Eq. (17) and the latter in Eq. (18).

D Rewriting the Projector

To prove Eq. (24), we first restate the definition of \mathcal{Z} from Eq. (25)

$$\mathcal{Z}(\rho) = \frac{1}{2^n} \sum_{k \in \{0,1\}^n} Z^{k_1} \otimes \dots \otimes Z^{k_n} \rho Z^{k_1} \otimes \dots \otimes Z^{k_n} \quad (43)$$

for an n -qubit state ρ . For a single-qubit state ρ_1 we note that

$$\sum_{k \in \{0,1\}} Z^k \rho_1 Z^k = 2 \sum_{j \in \{0,1\}} |j\rangle\langle j| \rho_1 |j\rangle\langle j| \quad (44)$$

by substituting $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$ as well as $\mathbb{I} = |0\rangle\langle 0| + |1\rangle\langle 1|$ into the left-hand side of Eq. (44). Consequently, by applying Eq. (44) iteratively, Eq. (43) becomes

$$\mathcal{Z}(\rho) = \sum_{j \in \{0,1\}^n} |j\rangle\langle j| \rho |j\rangle\langle j|. \quad (45)$$

Finally, we rewrite Eq. (45) as

$$\mathcal{Z}(\rho) = 2|1\dots 1\rangle\langle 1\dots 1| \rho |1\dots 1\rangle\langle 1\dots 1| \quad (46)$$

$$+ \sum_{j \in \{0,1\}^n} \beta_j |j\rangle\langle j| \rho |j\rangle\langle j| \quad (47)$$

$$= 2\mathcal{P}_{1\dots 1}(\rho) + \mathcal{P}(\rho) \quad (48)$$

where $\beta_j = -1$ for $j = (1, \dots, 1)$ and $\beta_j = 1$ otherwise, which proves Eq. (24).

E Derivation of sample-complexity overhead

This appendix investigates in more detail the sampling overhead associated with evaluating a cut circuit, closely following Refs. [24, 7]. In Appendix E.1 we prove the statements on sampling overhead made in the main text for the case where in each experimental run a circuit from the decomposition is sampled from a probability distribution. Subsequently, in Appendix E.2 we bound the variance by pre-estimating the expectation values of the partitioned circuits.

E.1 Sample complexity for circuit sampling

Consider an n -qubit quantum circuit. Assume the initial state $|0\rangle^{\otimes n}$ and consider the post-processing function $f : \{0, 1\}^n \rightarrow [-1, 1]$ on the measured bitstring s . Furthermore, assume that $f(s)$ can be efficiently calculated classically. These definitions describe the computational model of Refs. [42, 7]. The post-processing function gives rise to an observable O via

$$O = \sum_s f(s) P_s \quad (49)$$

where P_s is the projector on the computational basis state corresponding to bitstring s . Conversely, for instance, if the observable is a Pauli string, it can be written in the above form by local diagonalization and viewing the unitary diagonalization matrix as part of the circuit. The goal of the quantum computation is to approximate $\langle O \rangle$ to additive error ϵ with high probability for which we will define statistical estimators in the following.

Cutting a gate amounts to replacing its superoperator by a decomposition as in Eq. (1). We first consider the case where all \mathcal{F}_i in the decomposition correspond to unitary gates. With this substitution, we find

$$\langle O \rangle = \sum_i a_i \langle O \rangle_i = \sum_{i,s} p(s|i) a_i f(s) \quad (50)$$

where $\langle \cdot \rangle_i$ denotes the expectation value with respect to the state evolved by circuit i , containing \mathcal{F}_i . Moreover, $p(s|i)$ is the probability to measure bitstring s given circuit i . To evaluate the result on a quantum device, we sample i at each experimental run according to the probability distribution

$$p(i) = |a_i|/\kappa. \quad (51)$$

We therefore define a random variable I that takes values i with probability $p(i)$ and the estimator [25]

$$\hat{f} = \kappa \text{sign}(a_I) f(S_I) \quad (52)$$

where the random variable $S_{I=i}$ models the bitstring outcomes of the circuit i . This estimator is unbiased since

$$\mathbb{E}(\hat{f}) = \sum_{i,s} p(s,i) \kappa \text{sign}(a_i) f(s) \quad (53)$$

$$= \sum_{i,s} \kappa \text{sign}(a_i) p(i) p(s|i) f(s) \quad (54)$$

$$= \sum_{i,s} a_i p(s|i) f(s) \quad (55)$$

$$= \langle O \rangle. \quad (56)$$

Here, we substituted Eq. (51) into Eq. (54). The final result is obtained by comparing Eq. (55) to Eq. (50). The estimator for N shots is given by the sample mean of Eq. (52). Thus, to estimate $\langle O \rangle$ to additive error ϵ with probability $1 - \delta$, Hoeffding's inequality provides the required number of experimental repetitions as

$$N \geq 2 \frac{\kappa^2}{\epsilon^2} \ln \left(\frac{2}{\delta} \right) \quad (57)$$

where we used that $|\hat{f}| \leq \kappa$. The number of samples needed for the original circuit is obtained for $\kappa = 1$ in Eq. (57) from which we infer the sampling overhead $\mathcal{O}(\kappa^2)$.

It is straightforward to generalize this derivation to K cuts. In this case, the expectation value of the observable is obtained as

$$\langle O \rangle = \sum_{i_1, \dots, i_K} \sum_s p(s|i_1, \dots, i_K) a_{i_1} \cdot \dots \cdot a_{i_K} f(s) \quad (58)$$

and the estimator for K cuts becomes

$$\hat{f} = \kappa^K \text{sign}(a_{I_1}) \cdot \dots \cdot \text{sign}(a_{I_K}) f(S_{I_1, \dots, I_K}) \quad (59)$$

where we defined the independent random variables I_1, \dots, I_K that determine the specific circuit to run.

To show equality between the expectation value of Eq. (59) and Eq. (58), we follow Eq. (53) to Eq. (56) and make use of the independence of I_1, \dots, I_K . Since $|\hat{f}| \leq \kappa^K$, Hoeffding's inequality applied to the sample mean of Eq. (59), provides the bound $\mathcal{O}(\kappa^{2K})$ on the sampling overhead. If the observable factorizes over the partitions A and B of the original circuit, the post-processing function factorizes as well, that is $f(s) = f^A(s^A) f^B(s^B)$ where s^A and s^B are the bitstring results on partition A and B . If the initial state factorizes and all gates connecting the two partitions are cut, the partitioned circuits can be evaluated on independent quantum computers or sequentially on the same device.

We now turn to a quantum circuit that contains projectors. Consider the map \mathcal{M} consisting of a complete set of projectors P_l and $\xi_l \in [-1, 1]$ with

$$\mathcal{M}(\rho) = \sum_l \xi_l P_l \rho P_l \quad (60)$$

for state ρ . This map is neither positive since ξ_l can be smaller than zero, nor trace preserving since the state in Eq. (60) is not normalized after the projection and multiplied by ξ_l . Note that both $\mathcal{P}_{1\dots 1}$ and \mathcal{P} of Eq. (24) are instances of \mathcal{M} . For $\mathcal{P}_{1\dots 1}$ we set $\xi_l = 1$ only for $l = (1, \dots, 1)$ and zero otherwise. On the other hand for \mathcal{P} we have $\xi_l = -1$ for $l = (1, \dots, 1)$ and $\xi_l = 1$ otherwise. Even though \mathcal{M} does not correspond to a physical time evolution, we can nevertheless estimate

$$\chi = \text{tr}(OU \circ \mathcal{M}(\rho)) \quad (61)$$

by repeated use of a quantum computer. In Eq. (61), the density matrix ρ is the state right before the projectors and U is the unitary channel corresponding to the unitary evolution U afterwards until the final measurement. To estimate χ on a quantum computer, we perform intermediate measurements, and sample according to the estimator

$$\hat{f} = \xi_L f(S) \quad (62)$$

where the random variable S describes the bitstring outputs of the final measurements as before and the random variable L models the outcomes of the intermediate measurements. According to Eq. (62) we have to add $-f(S)$ in the sample mean for \hat{f} if we found the all-one state in the intermediate measurement and $f(S)$ otherwise. This estimator is unbiased since

$$\mathbb{E}(\hat{f}) = \sum_{s,l} p(s,l) \xi_l f(s) \quad (63)$$

$$= \sum_{s,l} \xi_l \text{tr}(P_s U P_l \rho P_l U^\dagger) f(s) \quad (64)$$

$$= \sum_s \text{tr}(P_s U \circ \mathcal{M}(\rho)) f(s) \quad (65)$$

$$= \chi. \quad (66)$$

The first line, Eq. (63), is the definition of the expectation value and in Eq. (64) we substituted $p(s, l) = \text{tr}(P_s U P_l \rho P_l U^\dagger)$ [1]. In the final step, we identified $O = \sum_s f(s) P_s$. The number of samples required to determine χ to given additive error is again determined by Hoeffding's inequality applied to the sample mean of Eq. (62). Since $|\xi_l| \leq 1$ by definition, Eq. (57) remains unchanged when some of the circuits contain projectors.

E.2 Sampling complexity by pre-estimating circuits

In Sec. 5, we show experimental results for a circuit that disintegrates into two independent partitions A and B . After cutting the single gate that connects the two partitions, we have to evaluate

$$\langle O \rangle = \sum_i a_i \langle O^A \rangle_i \langle O^B \rangle_i. \quad (67)$$

Rather than sampling a circuit pair i for each experimental run as discussed in Appendix E.1 we pre-estimate the expectation values $\langle O^A \rangle_i$ and $\langle O^B \rangle_i$ for all the circuits first and subsequently restore the result of the original circuit with the help of Eq. (67). In the following, it is shown that the standard deviation of Eq. (67) can be bounded by ϵ for a total number of experimental runs

$$N \geq 4 \frac{\kappa^2}{\epsilon^2} \quad (68)$$

where we allocate

$$N_i = \frac{1}{2} \frac{|a_i|}{\kappa} N \quad (69)$$

samples to each circuit of circuit pair i to determine the expectation values $\langle O^A \rangle_i$ and $\langle O^B \rangle_i$. Note that $2 \sum_i N_i = N$. Next, we define the estimator

$$\hat{f} = \sum_i a_i \hat{f}_i^A \hat{f}_i^B \quad (70)$$

where $\mathbb{E}(\hat{f}_i^A) = \langle O^A \rangle_i$ and $\mathbb{E}(\hat{f}_i^B) = \langle O^B \rangle_i$. Since \hat{f}_i^A and \hat{f}_i^B are independent, this estimator is unbiased. The variance of \hat{f} is calculated as

$$\text{Var}(\hat{f}_i^A \hat{f}_i^B) \quad (71)$$

$$= \text{Var}(\hat{f}_i^A) \mathbb{E}[(\hat{f}_i^B)^2] + \text{Var}(\hat{f}_i^B) [\mathbb{E}(\hat{f}_i^A)]^2 \quad (72)$$

$$\leq \text{Var}(\hat{f}_i^A) + \text{Var}(\hat{f}_i^B) \quad (73)$$

$$\leq \frac{2}{N_i} \quad (74)$$

In Eq. (71) we first factorized the variance, valid for independent random variables. Eq. (73) makes use of the bounds $\mathbb{E}[(\hat{f}_i^B)^2] \leq 1$ and $|\mathbb{E}(\hat{f}_i^A)| \leq 1$ since $|f_i^A| \leq 1$ and $|f_i^B| \leq 1$. Finally, $\text{Var}(\hat{f}_i) \leq 1/N_i$ for

N_i independent samples. Consequently,

$$\text{Var}(\hat{f}) = \sum_i a_i^2 \text{Var}(\hat{f}_i^A \hat{f}_i^B) \quad (75)$$

$$\leq 2 \sum_i a_i^2 / N_i \quad (76)$$

$$\leq \epsilon^2 \quad (77)$$

using Eq. (70), Eq. (74) and Eq. (69).

References

- [1] M. A. Nielsen and I. L. Chuang. "Quantum computation and quantum information: 10th anniversary edition". Cambridge University Press. (2010).
- [2] J. Preskill. "Quantum Computing in the NISQ era and beyond". *Quantum* **2**, 79 (2018).
- [3] P.W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In Proceedings 35th Annual Symposium on Foundations of Computer Science. Page 124. IEEE Comput. Soc. Press (1994).
- [4] L. K. Grover. "A fast quantum mechanical algorithm for database search". In Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing. Page 212. New York, NY, USA (1996). Association for Computing Machinery.
- [5] E. Farhi, J. Goldstone, and S. Gutmann. "A quantum approximate optimization algorithm" (2014). [arXiv:1411.4028](https://arxiv.org/abs/1411.4028).
- [6] "IBM Development Roadmap 2022". <https://www.ibm.com/quantum/roadmap>. Accessed: 2023-01-02.
- [7] T. Peng, A. W. Harrow, M. Ozols, and X. Wu. "Simulating large quantum circuits on a small quantum computer". *Phys. Rev. Lett.* **125**, 150504 (2020).
- [8] M. A. Perlin, Z. H. Saleem, M. Suchara, and J. C. Osborn. "Quantum circuit cutting with maximum-likelihood tomography". *NPJ Quantum Inf.* **7**, 64 (2021).
- [9] G. Uchihara, T. M. Aamodt, and O. Di Matteo. "Rotation-inspired circuit cut optimization". In 2022 IEEE/ACM Third International Workshop on Quantum Computing Software (QCS). Page 50. Los Alamitos, CA, USA (2022). IEEE Computer Society.
- [10] D. Chen, B. Baheri, V. Chaudhary, Q. Guan, N. Xie, and S. Xu. "Approximate quantum circuit reconstruction". In 2022 IEEE International Conference on Quantum Computing and Engineering (QCE). Page 509. (2022).
- [11] C. Ying, B. Cheng, Y. Zhao, H.-L. Huang, Y.-N. Zhang, M. Gong, Y. Wu, S. Wang, F. Liang, J. Lin, Y. Xu, H. Deng, H. Rong, C.-Z. Peng, M.-H. Yung, X. Zhu, and J.-W. Pan. "Experimental

- simulation of larger quantum circuits with fewer superconducting qubits”. *Phys. Rev. Lett.* **130**, 110601 (2023).
- [12] T. Ayril, F.-M. Le Régent, Z. Saleem, Y. Alexeev, and M. Suchara. “Quantum divide and compute: Hardware demonstrations and noisy simulations”. In 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Page 138. (2020).
- [13] T. Ayril, F.-M. Le Régent, Z. Saleem, Y. Alexeev, and M. Suchara. “Quantum divide and compute: Exploring the effect of different noise sources”. *SN comput. sci.* **2**, 132 (2021).
- [14] J. Liu, A. Gonzales, and Z. H. Saleem. “Classical simulators as quantum error mitigators via circuit cutting” (2022). [arXiv:2212.07335](https://arxiv.org/abs/2212.07335).
- [15] R. Majumdar and C. J. Wood. “Error mitigated quantum circuit cutting” (2022). [arXiv:2211.13431](https://arxiv.org/abs/2211.13431).
- [16] W. Tang, T. Tomesh, M. Suchara, J. Larson, and M. Martonosi. “CutQC: Using small quantum computers for large quantum circuit evaluations”. In Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems. Page 473. ASPLOS 2021 New York, NY, USA (2021). Association for Computing Machinery.
- [17] W. Tang and M. Martonosi. “ScaleQC: A scalable framework for hybrid computation on quantum and classical processors” (2022). [arXiv:2207.00933](https://arxiv.org/abs/2207.00933).
- [18] T. Chatterjee, A. Das, S. I. Mohtashim, A. Saha, and A. Chakrabarti. “Qurzon: A prototype for a divide and conquer-based quantum compiler for distributed quantum systems”. *SN Computer Science* **3**, 323 (2022).
- [19] K. Mitarai and K. Fujii. “Methodology for replacing indirect measurements with direct measurements”. *Phys. Rev. Research* **1**, 013006 (2019).
- [20] K. Mitarai and K. Fujii. “Constructing a virtual two-qubit gate by sampling single-qubit operations”. *New J. Phys.* **23**, 023021 (2021).
- [21] K. Mitarai and K. Fujii. “Overhead for simulating a non-local channel with local channels by quasiprobability sampling”. *Quantum* **5**, 388 (2021).
- [22] K. Temme, S. Bravyi, and J. M. Gambetta. “Error mitigation for short-depth quantum circuits”. *Phys. Rev. Lett.* **119**, 180509 (2017).
- [23] C. Piveteau, D. Sutter, and S. Woerner. “Quasiprobability decompositions with reduced sampling overhead”. *NPJ Quantum Inf.* **8**, 12 (2022).
- [24] C. Piveteau. “Advanced methods for quasiprobabilistic quantum error mitigation”. Master’s thesis. ETH Zurich. (2021). [url: https://www.research-collection.ethz.ch/handle/20.500.11850/504508](https://www.research-collection.ethz.ch/handle/20.500.11850/504508).
- [25] H. Pashayan, J. J. Wallman, and S. D. Bartlett. “Estimating outcome probabilities of quantum circuits using quasiprobabilities”. *Phys. Rev. Lett.* **115**, 070501 (2015).
- [26] C. Piveteau and D. Sutter. “Circuit knitting with classical communication” (2022). [arXiv:2205.00016](https://arxiv.org/abs/2205.00016).
- [27] G. Vidal and R. Tarrach. “Robustness of entanglement”. *Phys. Rev. A* **59**, 141 (1999).
- [28] A. Lowe, M. Medvidović, A. Hayes, L. J. O’Riordan, T. R. Bromley, J. M. Arrazola, and N. Killoran. “Fast quantum circuit cutting with randomized measurements”. *Quantum* **7**, 934 (2023).
- [29] Z. H. Saleem, T. Tomesh, M. A. Perlin, P. Gokhale, and M. Suchara. “Divide and conquer for combinatorial optimization and distributed computing” (2021). [arXiv:2107.07532](https://arxiv.org/abs/2107.07532).
- [30] S. Hadfield. “Quantum algorithms for scientific computing and approximate optimization” (2018). [arXiv:1805.03265](https://arxiv.org/abs/1805.03265).
- [31] T. Yamamoto and R. Ohira. “Error suppression by a virtual two-qubit gate”. *J. Appl. Phys.* **133**, 174401 (2023).
- [32] S. C. Marshall, C. Gyurik, and V. Dunjko. “High Dimensional Quantum Machine Learning With Small Quantum Computers”. *Quantum* **7**, 1078 (2023).
- [33] B. Coecke and R. Duncan. “Interacting quantum observables: categorical algebra and diagrammatics”. *New J. Phys.* **13**, 043016 (2011).
- [34] B. Coecke and R. Duncan. “Interacting quantum observables”. In L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfssdóttir, and I. Walukiewicz, editors, Automata, Languages and Programming. Page 298. Berlin, Heidelberg (2008). Springer Berlin Heidelberg.
- [35] J. van de Wetering. “ZX-calculus for the working quantum computer scientist” (2020). [arXiv:2012.13966](https://arxiv.org/abs/2012.13966).
- [36] A. Kissinger and J. van de Wetering. “Simulating quantum circuits with zx-calculus reduced stabiliser decompositions”. *Quantum Sci. Technol.* **7**, 044001 (2022).
- [37] A. Kissinger, J. van de Wetering, and R. Vilmart. “Classical simulation of quantum circuits with partial and graphical stabiliser decompositions”. In F.s Le Gall and T. Morimae, editors, 17th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2022). Volume 232 of Leibniz International Proceedings in Informatics (LIPIcs), page 5:1. Dagstuhl, Germany (2022).
- [38] M. Backens and A. Kissinger. “ZH: A complete graphical calculus for quantum computations in-

volving classical non-linearity”. *Electron. Proc. Theor. Comput. Sci.* **287**, 23 (2019).

- [39] “van de Wetering, J.”. Private communication.
- [40] “IBM Quantum”. <https://quantum-computing.ibm.com/>. Accessed: 2023-01-23.
- [41] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. “Elementary gates for quantum computation”. *Phys. Rev. A* **52**,

- 3457 (1995).
- [42] S. Bravyi, G. Smith, and J. A. Smolin. “Trading classical and quantum computational resources”. *Phys. Rev. X* **6**, 021043 (2016).
- [43] “Github repository MCZ-gate cutting”. https://github.com/ChristianUfrecht/MCZgate_cutting.
- [44] E. van den Berg, Z. K. Mineev, and K. Temme. “Model-free readout-error mitigation for quantum expectation values”. *Phys. Rev. A* **105**, 032620 (2022).

Supplementary material

In this supplemental material, we provide more details on our simulations and experiments on the `ibmq_ehningen` system.

Random circuits

For our numerical simulations in Sec. V of the main article we generated a set of random circuits consisting of about on average 30 single-qubit R_X , R_Y and R_Z rotations and 10 CNOT gates. We generated circuits with 3, 4, or 5 qubits with one MCZ gate in the center, accordingly, of order 3, 4, or 5. Two instances of these circuits are shown in Fig. 6 and Fig. 7. In the 3-qubit circuits, one 1-qubit and one 2-qubit partition, and in the 5-qubit circuits one 2-qubit and one 3-qubit partition are only connected by the MCZ gate. For the numerical evaluation, we choose the Pauli $Z \otimes \dots \otimes Z$ string as observable. For generating the random circuits, we impose the constraint that the MCZ gate in the circuits should have a significant impact on the expectation value of the observable in order to be able to resolve its influence on the result with the available number of experimental runs. That is, the difference in the expectation value between a circuit with and without MCZ is chosen to be greater than 0.2. A proof-of-concept implementation of MCZ-gate cutting for arbitrary order of the gate is available on GitHub [43].

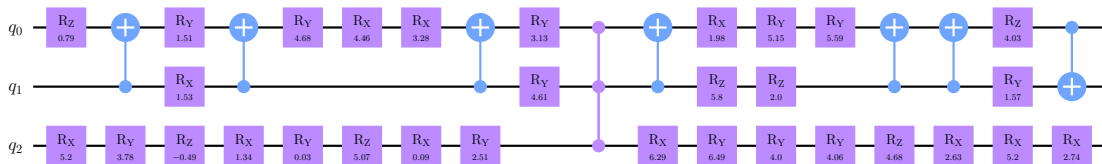


Figure 6: Random three-qubit circuit with a CCZ gate at the center.

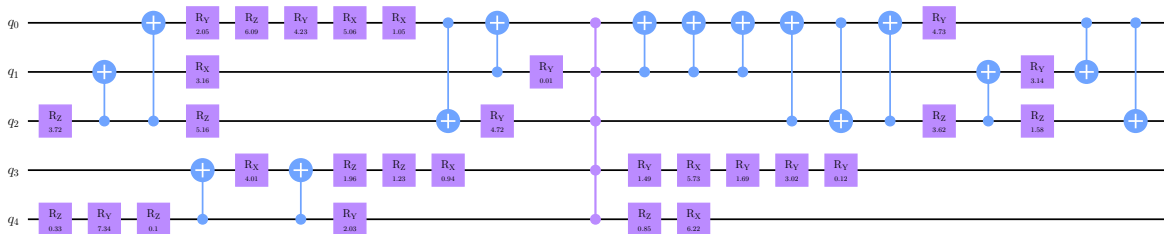


Figure 7: Random five-qubit circuit with an MCZ gate of order five at the center.

Experiments on `ibmq_ehningen`

In this part of the supplemental material, we provide more details on the experimental setup. For the experiments we (1) again generated a random 3-qubit circuit and a 5-qubit circuit with two partitions, only connected by a

CCZ gate and an MCZ gate of order 5, respectively. The circuits are shown in Fig. 8 and Fig. 9. Subsequently, we optimized the single-qubit rotation angles until the difference of the expectation values of a Pauli Z-string $Z \otimes \dots \otimes Z$ with respect to the circuit with and without the MCZ gate was maximal. Finally, we added one free parameter to the rotation angles of three single-qubit rotation gates, which were scanned in the experiments. The parameter is denoted by ψ in the circuits shown in Fig. 8 and Fig. 9. (2) We used the highest level of optimization offered by the Qiskit API after transpiling the circuits to the native gate set and the hardware graph of the `ibmq_ehningen` device. The highest level of optimization first searches for a layout that satisfies all the 2-qubit gate connectivity to the hardware graph considering the qubits readout errors and gate fidelities. Then the circuit is unrolled to the native gate set. Finally, optimizations in the form of commutative gate cancellation and re-synthesis of two-qubit unitary blocks are performed. (3) 18 data points between $[0, 2\pi]$ with equal intervals were chosen for scanning the free parameter. (4) We used 10^5 shots per data point, the maximum number of shots per job allowed by the `ibmq_ehningen` device. (5) All experiments were run with maximum error mitigation offered by `ibmq` wherever possible. We used the TREX option for readout-error mitigation [44] offered by Qiskit. (6) The `ibmq_ehningen` device can be characterized by the following parameters [40]: It is a 27 qubit `ibmq_falcon` processor, whose connectivity graph is shown in Fig. 10. The noise parameters at the time of our experiments were the following: Decoherence times: Average values $T_1 = 160 \mu\text{s}$ and $T_2 = 150 \mu\text{s}$ with large fluctuations between the qubits. Single-qubit errors are of the order of 10^{-4} and CNOT-gate errors of the order of 0.009.

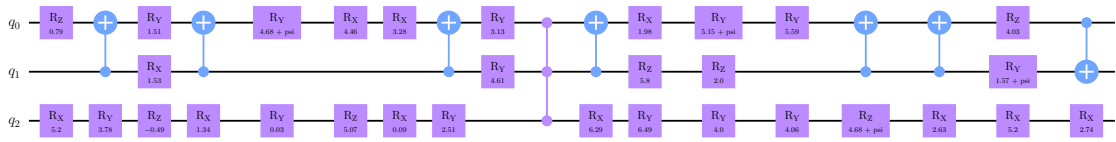


Figure 8: Circuit for the cutting of a CCZ gate.

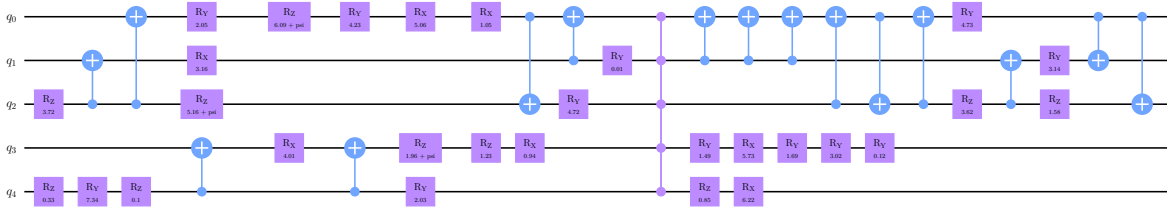


Figure 9: Circuit for the cutting of an MCZ gate of order 5.

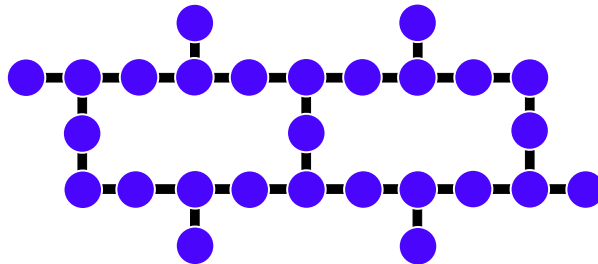


Figure 10: Connectivity graph of `ibmq_ehningen`.